

Adapting SDN datacenters to support Cloud IIoT applications

Pedro Gonçalves, Joaquim Ferreira
ESTGA/IT, Universidade de Aveiro
Aveiro, Portugal
{pasg, jjcf}@ua.pt

Paulo Pedreiras, Daniel Corujo
DETI/IT, Universidade de Aveiro
Aveiro, Portugal
{pbrp, dcorujo}@ua.pt

Abstract—IIoT (Industrial Internet of Things) cloud applications require reliable, fault-tolerant networks, supporting real-time guarantees and allowing interaction with other applications already existing in the datacenter. The Software Defined Networks (SDN) paradigm is especially suited for the management of the network cloud because of its fine grained admission control and the management flexibility provided by the centralized resource management.

This paper presents a SDN management approach for IIoT datacenters, which provides an efficient fault-tolerant network, enabled with deterministic QoS guarantees. By taking advantage of a centralized resource management mechanism, the proposed solution integrates topology definition with resource allocation, allowing to efficiently distribute available network resources for admitted packet flows.

Keywords— *SDN, Redundancy management, Resource management, Cloud datacenter*

I. INTRODUCTION

Industrial Internet is a relatively new area of research that aims to bring Internet technologies contributions for industrial processes, improving efficiency and flexibility of manufacturing equipment and reducing the cost of installation and maintenance of communication platforms. This approach assumes that manufacturing process sensors and actuators are connected by Ethernet-based systems to the factory management systems, which, in turn, are connected with more powerful data processing systems, usually housed in Cloud platforms. Those platforms usually apply virtualization technologies in order to adapt computing power demand, dynamically, by allocating virtual machines and network resources over the data centers' physical resources.

Software Defined Networks (SDN) present a new network management paradigm, creating a centralized resource management point, able to dynamically apply more complex network resource management policies and mechanisms. With SDN, network behavior becomes programmable according to a set of defined policies, improving network efficiency and safety, creating fault-tolerant networks, or even offering Quality of Service (QoS) guarantees to services. Fine grained and centralized network management approach brought SDN technology into cloud networks, allowing the integration of network and IT in the same management platforms [1].

Even though much work has been developed in recent years around SDN [2] [3] [4], both by academia and the industry,

solutions are usually sectorial, rarely addressing more than one particular issue at a time. Industrial Internet of Things (IIoT) networks require dependable network systems where industrial applications could truly rely on network resources to communicate. Such dependability must include efficient redundancy management of network resources and an effective implementation of QoS for the datacenter traffic implementing a resource management that could offer those guarantees.

This paper proposes a holistic management solution for network resources and redundancy, able to offer an efficient fault-tolerant network infrastructure, with QoS guarantees, for IIoT applications. Our proposal supports cohabitation with non-IIoT applications in the same datacenter, contributing as well with preliminary proof-of-concept results.

The rest of this paper is organized as follows: section 2 describes Industrial Internet environment, while section 3 presents some background on Software Defined Network technology. Section 4 presents the network management approach and describes some preliminary experimental results. Finally section 5 concludes the paper, pointing-out future lines of work.

II. IIoT NETWORKS ON INDUSTRIAL ENVIRONMENTS

The Internet of Things (IoT) [5] is a distributed computing paradigm based on the concept of pervasive communications between a large variety of things, like RFIDs, sensors and actuators, as well as the cloud computation platform located in data centers.

The IoT paradigm aggregated three complementary visions: the "Things"-oriented vision, enclosing wireless identification mechanisms such as RFID and NFC, everyday objects enabled with wireless sensors and actuators; the "Internet"-oriented vision including connectivity for everything and web of things approaches; and the "Semantic"-oriented vision, consisting in the use of semantic technologies to perform mining and analysis over massive amounts of data.

IoT solutions and proposals started to be deployed over everyday objects, as home automation solutions or precision agriculture, but then started to be considered as control solutions for industrial processes [6], as a means to aggregate Cyber-Physical Systems (CPS) [7] with Cloud computing facilities, that could run computing algorithms to empower industrial efficiency [8].

After the recent financial crisis, most countries focused their attention on the industry recovery and in improving their

efficiency through the use of ICT technologies in industrial environments, launching novel initiatives and programs. The Industrial Internet was firstly introduced in a 2012 [9], envisaging the integration of industrial machines with corresponding data systems using Big Data analytics techniques, allowing remote and centralized visualization, as well as the establishment of physical-human networks in order to ease the cooperation between humans and robots. *Industrie 4.0* [8] and *La nouvelle France industrielle* are two European examples of industry dynamics, in line with IIoT paradigm. Moreover, China proposed its “Made in China 2025” strategy to promote domestic integration of digital technologies and industrialization. High-level dialogue between the German and Chinese governments, on how the two manufacturing powerhouses could work together to accelerate the realization of the Industrial Internet in their two countries, has also been taking place. Several consortia, notably as the *Industrial Internet Consortium* (IIC), the *AllSeen Alliance* and *Open Interconnect Consortium* (OIC), emerged to address the growing need for collaboration on common concerns such as security and interoperability.

A. IIoT networks

IIoT networks follow, in general, the IoT network architecture and network elements. Physical and industrial processes are controlled through a set of sensors, actuators and cyber-physical systems, which are interconnected to the IP world via gateways.

The IP backbone connects the field level gateways, with local management systems and with the Internet world by means of Internet gateways. These allow the communication with other industrial facilities, as well as with Cloud platforms where smarter, powerful algorithms, associated with Big Data processing, take place.

The connection between CPS and Cloud platforms allows to vertically interconnect and embed production systems with economic processes, and to combine CPS systems horizontally in real-time networks.

Similar to the broader IoT market, the Industrial IoT market requires inexpensive nodes and communication technologies that are easy to work, install and manage. Given the characteristics of industrial processes, and the fact that these processes typically involve mechanical, physical and chemical safety-critical activities, industrial networks pose other specific requirements, mainly related with performance and reliability.

B. Industrial networks requirements

Industrial networks are tightly tied to CPS systems, coupling computational and physical elements, often forming a feedback loop where physical processes affect the computations and vice-versa. This tight coupling presents strict requirements in terms of predictability, latency, dependability and security.

Although both require the same reliability from the network, computational processes existing in the cloud and control applications from industrial plants have different communication requirements. Industrial plant IP backbone applications require strict timeliness, in contrast with generic cloud network applications.

III. SOFTWARE DEFINED NETWORKS

SDN [1] is a network management paradigm born in campuses networks, which soon spread to other environments. This management paradigm decouples the network control system (control plane) from the underlying network hardware, responsible for forwarding network packets to their destination (data plane). OpenFlow [10] is the most popular SDN technology which provides a standard API to communicate with network equipment and instructs it how to behave upon certain traffic classes/patterns, via the Controller entity. Under this scheme, network configuration is centralized and greatly simplified, and complex protocols traditionally used to shape the network operation and management can now traverse the network with a lean programming environment, following a *softwarization* of networking procedures approach.

A vast amount of work has been carried out on the Controller side [11][12], finding new schemes to organize and simplify the northbound API that abstracts SDN details to network applications.

Amongst other reasons, SDN is being adopted as a mean to achieve hierarchical virtualization of network resources [13]. Nevertheless, SDN based solutions for Ethernet resource management [14][15] are usually sectorial, just addressing one management issue a time. For example, in [16] a proactive fault tolerant network management solution has been developed, making use of alternative routes and a mechanism to notify link breakdown events, delivering network packets even under random link failure conditions. Despite effectively solving the fault-tolerance issues, the work did not consider any aspects related to the rational use of network resources nor with QoS mechanisms.

In [3] a reactive fault-tolerant resource management scheme for fat-tree [17] data center networks is presented, using flat IP addressing. The solution uses multipath load balancing techniques, but doesn't consider QoS requirements, absolutely necessary in industrial scenarios. Moreover, the associated results present a considerably high recovery time, which is not fast enough to maintain TCP sessions over link failures. Similarly, [4] presents a reactive fault-tolerant management solution that performs link failure detection and new topology establishment for fat-tree data center networks, but it does not consider multipath nor network resource optimization. Its results obtained in *Mininet* emulation tests, show even worse network restoration times, which are associated to Floodlight [14] controller.

Parallel to this, the cohabitation of different applications, with different network requirements, such as the applications present in data centers, represents a huge traffic engineering challenge that requires a deep awareness of each application traffic profile. Unable to perform such a thorough traffic analysis, network administrators implement a protection measure consisting in overprovisioning network resources, maintaining a comfortable margin of free resources in the hope that they will be enough to transport information at unknown peak traffic conditions. However, this type of practice does not favor anyone truly: the data center customer has no quality assurance for the contracted services, and the data center

operator has a huge waste of network resources and energy consumption.

The inefficient use of network resources in the data center is reflected also in the selection of routes used by data streams, usually determined by the selection of the network paths between the source and the destination. Standard approaches usually culminate in the overloading of the shortest paths. As a result, resource management algorithms have to include load-balancing mechanisms.

IV. A DEPENDABLE SDN NETWORK FOR IIOT CLOUDS

In order to take full advantage of SDN network management, we based our development on a multi-connected network architecture following a fat-tree topology (Fig. 1), where each node is enabled with a direct link to the SDN controller.

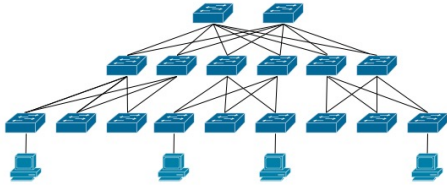


Fig. 1 – Reference topology

According to our model, the SDN controller is responsible for the topological management, admission control and resource allocation over the network resources of the SDN switches under its control. Given that any of the previously referred management actions have an effect on the success of the remaining controller processes, they are executed in a cyclical manner (Fig. 2) until a successful execution of the complete management cycle, or the controller has tested all combinations of the existing topologies, without having successfully allocated the flows.

Our management process could be triggered by a number of network events: a topological change in the network (link-down or a link-up event); the appearance of a new flow, or just the need for allocating previously admitted flows over the existing network resources.

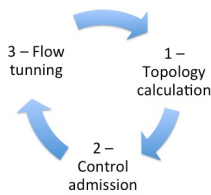


Fig. 2 - Controller management process.

A. Topology calculation

With the increasing reliance on cloud-based and virtualized services, there is still a lack of solutions to provide enough redundancy and explore all functionalities, aiming to 100% uptime. Despite the evolution of Ethernet redundancy mechanisms, such as RSTP, TRILL and SPB, or proprietary protocols such as QFabric and Virtual Cluster Switching (VCS), they still create a tree-based network topology, and there is still not a full use of all the equipment deployed on the network, contributing to a lack of overall efficiency in

resources usage. Moreover, traffic transfer between redundant links is still handled by a single node, the root bridge, a bottleneck and a single point of failure.

Our SDN-based network topology proposal allows computing, from a physically redundant network, the logical network topology that best connects each pair of source and destination nodes. Moreover, and in order to obtain a redundant logical network offer, the SDN controller performs two roles: it defines a primary path, and a backup path for the flows; and it instructs network switches to detect and to notify topology changes.

The path calculation between points is performed in two phases; firstly the shortest path between the two points using a *Dijkstra's* algorithm is computed, thus establishing the primary path. Afterwards, the edges belonging to the primary path from the network graph are removed and *Dijkstra's* algorithm is executed again in order to obtain a completely independent backup path.

The notification of a network topology change (Fig. 3) is also implemented in two phases. In the first phase, network switches notify the SDN controller (4) that the network topology was changed, asking it to establish a new network topology. In the second phase, switches reroute received packets over the backup route (5), thus triggering neighbor switches' learning process on changed topologies. The existence of backup routes allows the network to continue to operate using backup resources until the controller completes the topology definition (8) on the network switches.

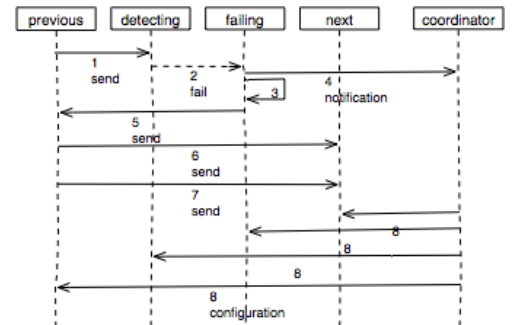


Fig. 3 – Topology notification

B. Admission control

Similar to any admission control process, SDN compares flow requirements with the existing network resources. Contrary to the non-redundant IP network admission controllers, our SDN controller iterates its admission control process over the complete list of network routes. Our admission control framework exploits existing mechanisms from the OpenDayLight project [18]. In order to ease the parameterization of the admission control module, QoS policies are currently being defined and a QoS parameterization interface will be implemented soon.

If the listed routes could support the resource allocation of the new flow, the flow is admitted, otherwise it is discarded. Additionally switches are configured to policy traffic flows in order to protect real-time flows from non-real-time network over usage.

C. Allocation optimization

After a successful admission of a new flow, a new step has to be taken by the controller, optimizing network resource allocation and allocating the required flow resources. Our resource management approach is also coupled with per-flow load balancing mechanisms, integrating them in the resource management solution, triggered by the predictive fault-tolerant mechanism. The balancing is performed in a per-flow basis in order to avoid TCP session throughput degradation, due to out of order packet arrival in case of per-packet balancing.

V. PRELIMINARY RESULTS

Our resource management solution is still a work in progress, so the results are still partial and preliminary. It was based on the development of an *OpenDayLight* controller application, that establishes a dependable network topology, enabled with proactive redundant paths [16].

A set of functional tests was developed by emulating in *Mininet* a network enabled with redundant connection (Fig. 1) and managed by the SDN controller. Ping sessions between each station were established and random node failures were emulated. Network traffic was captured, in order to analyze packet drops and packet delivery delays. Additionally, a scalability assessment of the management solution was performed, evaluating its applicability to a real datacenter scenario.

A. Experimental results

Tests results shown zero packet loss upon topology reconfiguration, and packets were redirected through backup paths to their destination whenever changes in the topology provided a disruption in the main path, in less than 350 ms, independently of the time needed by the controller for calculating a new topology.

Despite *Mininet* emulation limitations that reduced the number of independent streams to 600 in the switched network, scalability tests showed very promising results with respect to the applicability of such management approach to a real datacenter scenario. Moreover, it was observed that packet delivery delays are highly dependent on the number of flows that have to be migrated to new links, independently of the absolute number of flows transported by the datacenter network.

VI. CONCLUSIONS

This paper has described work in progress on the development of a network management platform for an IIoT cloud. The management approach considers a tree-phase process to compute a network topology: admission control of network flows, optimization of network resource allocation and network load balancing.

Preliminary results allow us to confirm that the network supports random failures with no packet loss, by using an alternative route mechanism, and allows us to expect a scalable solution appropriated to manage a real datacenter network.

Plans for future work include completing the development of the management solution and evaluating the effect of

topological changes in the resource allocation of admitted flows. We also plan to study the dynamic behavior of the network upon changes, e.g., due to link or switch failures, of the network topology.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7) under grant agreement n. 3176711.

REFERENCES

- [1] A. Leon-Garcia, H. Bannazadeh, and Q. Zhang, "Openflow and SDN for Clouds," in *Cloud Services, Networking, and Management*, John Wiley & Sons, Inc, 2015, pp. 129–152.
- [2] E. Jo, D. Pan, J. Liu, and L. Butler, "A Simulation and Emulation Study of SDN-based Multipath Routing for Fat-tree Data Center Networks," in *Proceedings of the 2014 Winter Simulation Conference*, 2014, pp. 3072–3083.
- [3] R. M. Ramos, M. Martinello, and C. E. Rothenberg, "Data Center Fault-Tolerant Routing and Forwarding: An Approach Based on Encoded Paths," *Dependable Computing (LADC)*, 2013 Sixth Latin-American Symposium on, pp. 104–113, 2013.
- [4] J. Li, J. Hyun, J.-H. Yoo, S. Baik, and J. W.-K. Hong, "Scalable failover method for Data Center Networks using OpenFlow," *Network Operations and Management Symposium (NOMS)*, 2014 IEEE, pp. 1–6, 2014.
- [5] N. Gershenfeld, R. Krikorian, and D. Cohen, "The Internet of things.," *Sci. Am.*, vol. 291, pp. 76–81, 2004.
- [6] "Industrial Internet of Things: Unleashing the Potential of Connected Products and Services," 2015.
- [7] E. A. Lee, "Cyber Physical Systems: Design Challenges," in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, 2008, pp. 363–369.
- [8] R. Drath and A. Horch, "Industrie 4.0: Hit or hype? [Industry Forum]," *IEEE Industrial Electronics Magazine*, vol. 8, pp. 56–58, 2014.
- [9] P. C. Evans and M. Annunziata, "Industrial Internet: pushing the boundaries of minds and machines," 2012.
- [10] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling Innovation in Campus Networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, p. 69, Mar. 2008.
- [11] D. Kreutz and F. Ramos, "Software-Defined Networking: A Comprehensive Survey," *arXiv Prepr. arXiv* ..., p. 49, 2014.
- [12] B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *IEEE Communications Surveys and Tutorials*, 2014.
- [13] J. Matias, B. Tornero, A. Mendiola, E. Jacob, and N. Toledo, "Implementing Layer 2 Network Virtualization Using OpenFlow: Challenges and Solutions," *2012 Eur. Work. Softw. Defin. Netw.*, vol. 2, pp. 30–35, Oct. 2012.
- [14] R. Wallner and R. Cannistra, "An SDN Approach : Quality of Service using Big Switch 's Floodlight Open-source Controller," vol. Vol 35, pp. 14–19, 2013.
- [15] N. Blefari-Melazzi, A. Detti, G. Morabito, S. Salsano, and L. Veltri, "Information Centric Networking over SDN and OpenFlow: Architectural Aspects and Experiments on the OFELIA Testbed," *Arxiv Prepr.*, 2013.
- [16] P. Goncalves, A. Martins, D. Corujo, and R. Aguiar, "A fail-safe SDN bridging platform for cloud networks," in *Telecommunications Network Strategy and Planning Symposium (Networks)*, 2014 16th International, 2014, pp. 1–6.
- [17] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *Proceedings of the ACM SIGCOMM 2008 conference on Data communication - SIGCOMM '08*, 2008, pp. 63–74.
- [18] J. Medved, A. Tkacik, R. Varga, and K. Gray, "OpenDaylight: Towards a Model-Driven SDN Controller architecture," *A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2014 IEEE 15th International Symposium on, pp. 1–6, 2014.